Методичка 5.1 - Способы обхода активации программ

Обход активации программ

По сути, обход активации программы - это взлом программы.

Согласно википедии, взлом программного обеспечения - это действия, направленные на устранение защиты программного обеспечения, которая была встроена разработчиками для ограничения функциональных возможностей.

Процедура активация программы, это и есть защита программы. Таким образом, взлом сводится, либо к отключению такой защиты, либо к ее обходу.

Способы обхода активации программ

Для достижения этой цели может быть использованы различные подходы и способы. Далее будут рассмотрены следующие способы:

- Продление триального периода
- Создание ключа
- Патчинг
- Перенаправление сетевых запросов

Продление триального периода

Думаю, многие знают, что есть программы, которые имеют, так называемый, триальный период. Это такой период, в который программа не имеет ограничений по функциональности. Этот период может быть завязан, как на временной промежуток, например, 30 дней, так и на количество запусков, например, можно воспользоваться программой 10 раз, а на 11-й раз программа попросит ввести ключ.

Обычно, для подтверждения оплаты программа требует ввести ключ, который вы получаете, например, на почту после того, как вы купили программу.

Для того, чтобы программа могла понять, что триальный период истек, ей необходимо, либо при установке, либо при первом запуске создать контрольную точку, а потом регулярно с ней сверяться.

Важно понимать, что эта контрольная точка должна где-то находиться в системе. А если она есть, мы всегда можем ее либо удалить, либо изменить для того, чтобы продлить триальный период.

Основная сложность заключается в том, чтобы определить место, где программа создает контрольную точку. Это может быть что угодно, например, скрытый файл в домашней директории пользователя или параметр в ветке реестра Windows.

В некоторых случаях, для продления триального периода достаточно изменить дату в операционной системе, например, на 30 дней назад и программа будет работать еще 30 дней.

Так когда-то давно можно было обойти проверку для антивируса Касперского. В современных приложениях уже не всё так просто, но идея осталась прежней.

Продление триального периода

Для демонстрации того, как можно продлить триальный период нам понадобится программа, в которой будет реализована возможность использования программы в пробном режиме. Специально для этой цели была подготовлена программа prog-4.1, которая умеет складывать 2 числа. Но в бесплатном режиме позволяет воспользоваться программой только 10 раз.

Исходный код программы prog-4.1

```
#include <windows.h>
…
```

Компиляция:

```
> cl prog-4.1.c -Od /Gs- /GS- /link /DYNAMICBASE:NO /NXCOMPAT:NO
```

Запуск:

```
> prog-4.1.exe

> prog-4.1.exe 1 2

> prog-4.1.exe 5 6

> prog-4.1.exe 3 4
```

Мы видим, что триальный период заканчивается. Давайте попробуем узнать, где находится контрольная точка.

Для того, чтобы определить, где находится контрольная точка, можно воспользоваться утилитой Process Monitor.

Process Monitor позволяет отслеживать в режиме реального времени изменения в файловой системе и реестре Windows в рамках выбранного процесса.

Рекомендации по установке утилиты Process Monitor.

Ссылка: https://docs.microsoft.com/en-us/sysinternals/downloads/procmon

Открываем программу Process Monitor. Видим, что в данный момент в системе происходит очень много различных изменений. Давайте добавим фильтр, чтобы видеть изменения только по программе prog-4.1.

Выбираем фильтр по имени процесса - prog-4.1.exe. Давайте запустим еще несколько раз программу и посмотрим, что наша программа будет изменять в системе.

Просматриваем список изменений в системе, который был сделан программой и видим, что программа уменьшает значение Counter в одной из веток реестра.

Давайте отметим все изменения этого флага. Кликаем правой кнопкой - Highlights - Path.

Запустим программу еще несколько раз. Попытки закончились и программа больше не работает. Необходимо ввести ключ, но ключа у нас нет.

Видим, что счетчик флаг уменьшился до нуля. Давайте откроем редактор реестре - regedit и попробуем изменить значение флага.

HKEY_CURRENT_USER\Software\GeekBrains\Prog4

Выставляем значение 5 и пробуем запустить программу. Программа заработала. Мы научились сбрасывать триальный период для программы prog-4.1.

В некоторых случаях после анализа результатов утилиты Process Monitor становится ясно, где программа хранит контрольную точку, как это произошло для программы prog-4.1.exe, но бывают программы, которые имеют несколько процессов и делают сотни и тысячи изменений на диске при установке или старте.

В таких случаях без изучения внутреннего устройства программы не обойтись.

В целом, воздействие на программу путем продления триального периода является, как мне кажется, самым легким и эффективным способом, который позволяет пользоваться платной версией программы и при этом не платить за нее. С связи с этим, очень часто на пиратских сайтах можно найти специальные утилиты, которые обычно называются - Trial Reset. Они, как раз, сбрасывают триальный период для платный версии программы.

Создание ключа

Бывает так, что разработчики не предоставляют триального периода для платных версий программ. Есть бесплатная версия с ограниченным функционалом и платная, которая требует ключ при запуске.

В таких случаях выполняют реверс-инжиниринг программы, чтобы понять алгоритм проверки ключа. Знать алгоритм необходимо для того, чтобы составить свой ключ, который пройдет все проверки внутри программы.

Данный способ является самым сложным, но при этом самым стабильным, так как при вводе правильного ключа, мы получаем программу, которая будет работать также, как и та, что была куплена официально.

Патчинг

Бывает так, что алгоритм проверки восстановить крайне сложно или это требует очень много времени. В таких случаях может быть использован метод патчинга.

В переводе с английского Patch переводится, как заплатка. В данном случае, заплатка должна повлиять на логику работы программы, например, на процедуру проверки ключа.

Патчинг может быть выполнен, как для PE-файла, который хранится на жестком диске, так и для образа PE-файла в виртуальной памяти.

Ключевым отличием первого подхода является то, что патчинг PE-файла осуществляется до запуска программы и изменения вносятся на постоянной основе. Во втором случае модифицируется только виртуальная память, например, секция кода или данных.

Первый подход часто используется для взлома игр. Например, после установки игры, чтобы ключ лицензии не проверялся, на компьютере требуется заменить один EXE-файл на другой. В данном случае второй EXE-файл был предварительно пропатчен и не требует ключ при запуске игры.

Второй подход тоже может быть использован в играх, например, многим известная утилита ArtMoney, может во время игры прибавить количество жизней или количество денег.

Утилита ArtMoney подключается к существующему процессу игры, также как это делает, например, отладчик OllyDbg, и изменяет данные в виртуальной памяти.

Перенаправление сетевых запросов

Существуют программы, которые выполняют проверку ключа на удаленном сервере. В таком случае, реверс-инжиниринг программы никак не поможет для восстановлении алгоритма проверки ключа, так как этого алгоритма просто нет в программе.

Ключ, который вводится пользователем в программе, отправляется на удаленный сервер в сети Интернет и там проверяется. После проверки ключа программа получает ответ от удаленного сервера: true или false.

В таким случаях для обхода активации может быть использован один из следующих способов:

- Патчинг
- Перенаправление сетевых запросов

В случае патчинга алгоритм предельно простой - мы модифицируем программу так, чтобы запросы не отправлялись совсем.

Второй способ является более интересным и не требует модификации бинарного кода программы.

Для реализации этого способа необходимо создать собственный сервер проверки ключа и перенаправлять запросы на него. Он будет всегда отвечать так, как нужно нам.